DESARROLLO WEB
PHP

ACTIVIDAD 3

PRESENTADO POR: ENZO REYES H.

CONTENIDO



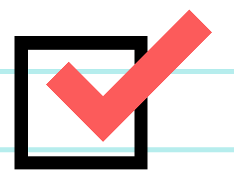
RAZONES PARA USAR FUNCIONES



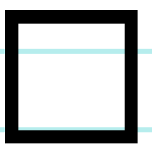
COMUNIDADES VIRTUALES DE AYUDA

SENA - CURSO PROGRAMACIÓN WEB PHP
CÓDIGO 1633069

DESARROLLO



RAZONES PARA USAR FUNCIONES



COMUNIDADES VIRTUALES DE AYUDA

IF ($HAYCAFE==TRUE){$TOMAR
}ELSE{$HACERCAFE}

SENA - CURSO PROGRAMACIÓN WEB PHP
CÓDIGO 1633069



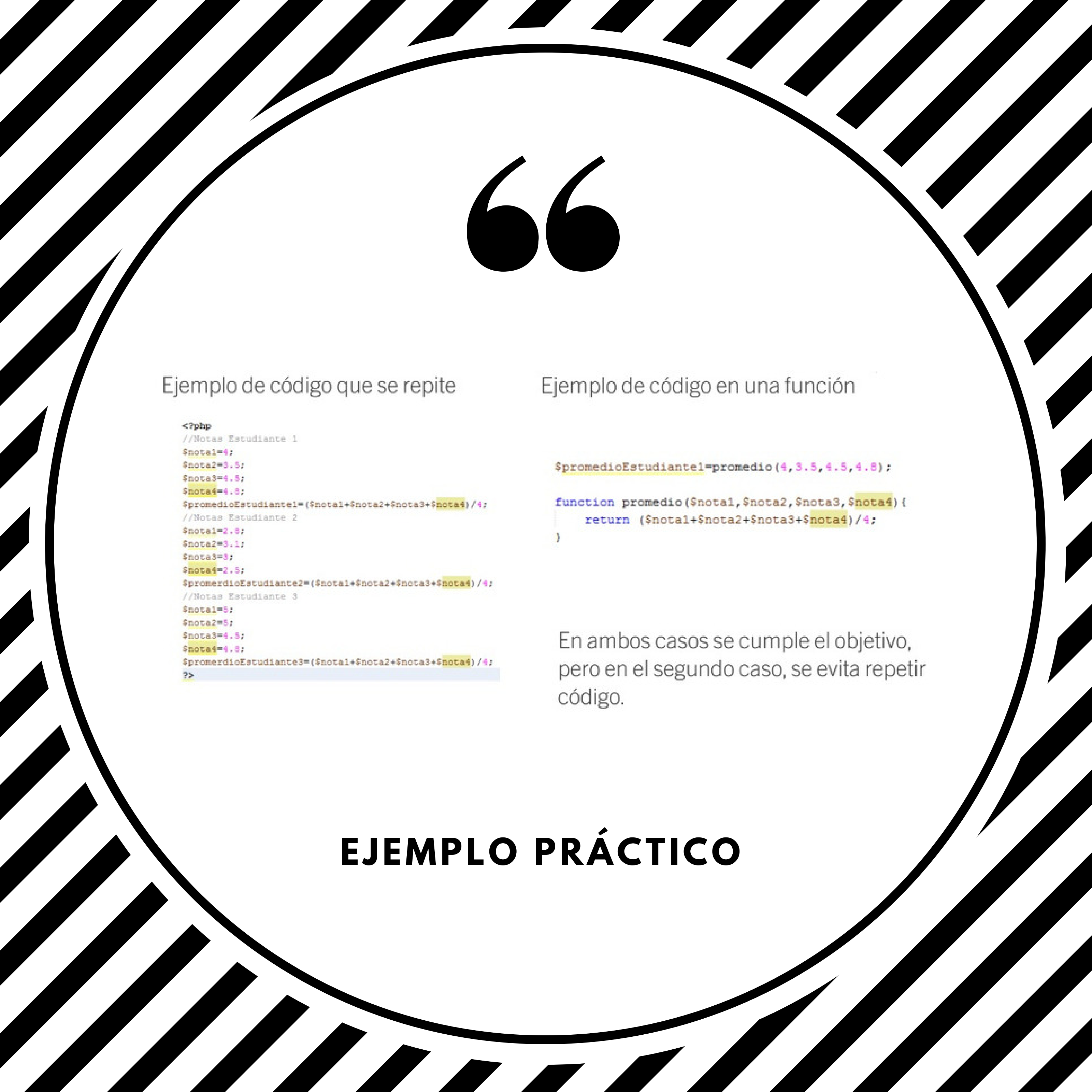
RAZÓN

01

SIMPLIFICAR EL CÓDIGO

ES DECIR:

Usar funciones nos evita el poner muchas veces la misma instrucción iterativa.


Ejemplo de código que se repite
<?php
//Notas Estudiante 1
$nota1=4;
$nota2=3.5;
$nota3=4.5;
$nota4=4.8;
$promedioEstudiante1=($nota1+$nota2+$nota3+$nota4)/4;
//Notas Estudiante 2
$nota1=2.8;
$nota2=3.1;
$nota3=3;
$nota4=2.5;
$promerdioEstudiante2=($nota1+$nota2+$nota3+$nota4)/4;
//Notas Estudiante 3
$nota1=5;
$nota2=5;
$nota3=4.5;
$nota4=4.8;
$promerdioEstudiante3=($nota1+$nota2+$nota3+$nota4)/4;
?>
Ejemplo de código en una función
$promedioEstudiante1=promedio(4,3.5,4.5,4.8);
function promedio($nota1,$nota2,$nota3,$nota4){
return ($nota1+$nota2+$nota3+$nota4)/4;
}
En ambos casos se cumple el objetivo, pero en el segundo caso, se evita repetir código.
EJEMPLO PRÁCTICO



RAZÓN 02
MODULARIDAD
Una función permite a nuestro código ser particionado para obtener límites bien definidos y por sobre todo mejora la comprensión del mismo.
Bertrand Meyer

“

```php
$promedioEstudiante1=promedio(4,3.5,4.5,4.8);

function promedio($nota1,$nota2,$nota3,$nota4){
    return ($nota1+$nota2+$nota3+$nota4)/4;
}
```

OTRO EJEMPLO

Si un valor en una función son números, por obligación los demás símiles serán números

ESTANDARIZACIÓN

RAZÓN 03

#FUNCIONES()

REUSO

RAZÓN 04

Al igual que este slide una función permite reutilizarse las veces que sean necesarias

RAZÓN 04

```
function promedio($nota1,$nota2,$nota3,$nota4){
    return ($nota1+$nota2+$nota3+$nota4)/4;
}

$promedioEstudiante1=promedio(4,3.5,4.5,4.8);

$promedioEstudiante2=promedio(2.8,3.1,3,2.5);

$promedioEstudiante3=promedio(5,5,4.5,4.8);
```

EJEMPLO: LA MISMA FUNCIÓN CON DIFERENTES VALORES

COLECCIONES

RAZÓN 05

TODAS LAS FUNCIONES QUE GENEREMOS LAS PODREMOS ALMACENAR EN UNA O MUCHAS BIBLIOTECAS PARA QUE PUEDAN SER INVOCADAS DESDE CUALQUIER PARTE DE NUESTRO PROYECTO POR MEDIO DEL ELEMENTO INCLUDE() O REQUIRE ()

```php
<?php

/*
 * Conjunto de funciones
 */

//Función para calcular el promedio
function promedio($nota1,$nota2,$nota3,$nota4)
{
    return ($nota1+$nota2+$nota3+$nota4)/4;
}

//Función para determinar el estado del estudiante de acuerdo al promedio
function estadoEstudiante($promedio)
{
    if($promedio>=3)
    {
        $estado="Aprobado";
    }
    else
    {
        $estado="Reprobado";
    }
    return $estado;
}
```

EJEMPLO: COLECCIÓN DE FUNCIONES

RAZÓN 06

SON FÁCILES DE EMPLEAR Y RÁPIDAS DE INVOCAR

Al momento de crearse se debe tener una buena abstracción para que ofrezca el resultado deseado, no obstante, una vez creada nuestra función es muy fácil invocarla en nuestro código

DEBE CREARSE CON UNA BUENA LÓGICA

```
function sumar($numero1, $numero2){
    echo "La suma es: ".($numero1 + $numero2);
}

//probando la funcion
sumar(2,6);
```

EJEMPLO DE LÓGICA E INVOCACIÓN DE UNA FUNCIÓN


RAZÓN 07
SON
GLOBALES



RAZÓN 07

AL SER ALMACENADAS EN BIBLIOTECAS PUEDEN VOLVERSE PLANTILLAS PARA FUTUROS PROYECTOS

ES POR ESO QUE SE PUEDEN VER LOS FAMOSOS FRAMEWORKS TAL COMO "SYMFONY"


index.php
Source
History
<?php
define('DS', DIRECTORY_SEPARATOR);
define('ROOT', realpath(dirname(__FILE__)) . DS);
define('APP_PATH', ROOT . 'application' . DS);
require_once APP_PATH . 'Config.php';
require_once APP_PATH . 'Request.php';
require_once APP_PATH . 'Bootstrap.php';
require_once APP_PATH . 'Controller.php';
require_once APP_PATH . 'Model.php';
require_once APP_PATH . 'View.php';
require_once APP_PATH . 'Registro.php';
echo '<pre>';print_r(get_required_files());
?>
EJEMPLO DE UN FRAMEWORK
BÁSICO

DESARROLLO





REQUIRE ("TOMARCAFE.PHP");
REQUIRE ("GALLETAS.PHP");

SENA - CURSO PROGRAMACIÓN WEB PHP
CÓDIGO 1633069



UNA COMUNIDAD HECHA POR DESARROLLADORES PARA DESARROLLADORES



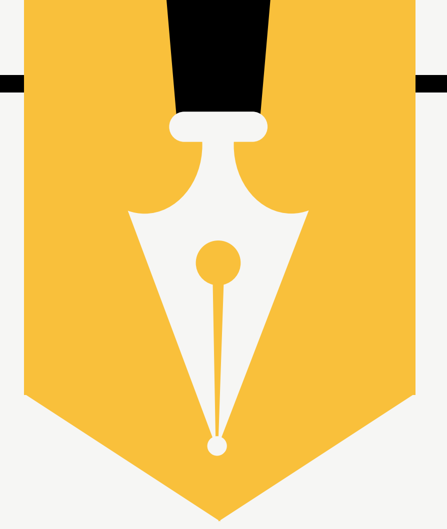
PROPÓSITO

ENTREGAR CONOCIMIENTO AL MUNDO

Esta es una comunidad hecha para compartir, departir, debatir, preguntar sobre diversos temas en programación bajo diversos lenguajes, la desventaja de esta comunidad es que la mayoría de su comunidad es de habla inglesa, no es apta para quienes no conocen del idioma.



UNA COMUNIDAD HECHA PARA SOLUCIONAR MUCHOS DE TUS PROBLEMAS



PROPÓSITO

DESARROLLAR PROYECTOS EN CONJUNTO

Esta comunidad a diferencia de la primera goza de una serie de repositorios de los cuales un desarrollador puede hechar mano a su vez, quienes desarrollan pueden subir sus ideas o trabajos para que puedan ser potenciados por otros miembros de la comunidad, ejemplo de ellos es Atom, el editor de código hackeable.

la desventaja es que para su uso ya se deben de tener bases de desarrollo para poder comprender y aprovechar sus recursos



UNA COMUNIDAD HECHA PARA COMPARTIR CONOCIMIENTO CON EL MUNDO



PROPÓSITO

COMPONER LA COMUNIDAD HISPANOHABLANTE MAS GRANDE EN DESARROLLO PHP

Ideada como una red social, esta comunidad permite al usuario crear sus entradas con preguntas o trozos de código para que sean revisados y mejorados, su desventaja es que al estar en maduración es probable que tus dudas no sean resueltas a la inmediatez.

¡MUCHAS GRACIAS!

ALGUNAS PARTES DE ESTE MATERIAL FUERON EXTRAÍDAS DE: HTTPS://APP.EMAZE.COM/USER/JMDIAZPU, WIKIPEDIA.COM, Y COMUNIDADES CONSULTADAS

EL CONOCIMIENTO ES PARA TODOS, POR LO CUAL:

